

REMARKS

SUMMARY

Claims 1-21 were rejected in the above-identified final Office Action. Claims 1, 11, and 21 have been amended. Accordingly, claims 1-21 are pending in the application.

Applicants appreciatively acknowledge the Examiner's consideration of and response to Applicants' arguments as presented in the Response filed on September 19, 2006.

CLAIM OBJECTIONS

In "Claim Objections," item 2 on page 2 of the above-identified final Office Action, claims 1, 11, and 21 are objected to because of an informality. More specifically, the Examiner notes that the phrase "unnested data processing cell" requires syntactic adjustment to be commensurate with the originally filed Specification.

In response, Applicants have amended the objected-to claims to adjust their scope. Claims 1, 11, and 21 now recite that the first and second data processing cell specifications are "unnested with respect to each other." Thus, Applicants now avoid the problems pointed out by the Examiner associated with interpreting unnested as "absolutely unnested." As is now recited, the first cell may not be nested within the second, and the second cell may not be nested within the first.

Accordingly, Applicants respectfully submit that the objections are now obviated.

CLAIM REJECTIONS UNDER 35 U.S.C. § 102

In "Claim Rejections – 35 USC § 102," item 4 on page 3 of the above-identified final Office Action, claims 1-6, 8-16, and 18-21 have been rejected as being fully anticipated by U.S. Patent Number 6,993,657 to *Renner et al.* (hereinafter "Renner") under 35 U.S.C. § 102(e).

Amended claim 1 recites a "method of computing comprising:

receiving at execution time, a data processing specification having a first and a second data processing cell specification, unnested with respect to each other, specifying a first and a second data processing cell respectively, with each data processing cell specification having a plurality of statements including a formula specifying an action or computation, the first data processing cell having a data dependency on the second data processing cell, and specified in a manner to be analyzed before the second data processing cell; analyzing in real time, the first and then the second data processing cell specification to determine execution order of the actions/computations specified by the first data processing cell specifications, based at least in part on interaction or computation references between the actions or computations specified; and effectuating the data processing specified by the data processing specification in accordance with the determined execution order of said actions/computations specified by said first and second data processing cell specifications.”

In contrast, Renner simply teaches a community server adapted to provide access and management of one or more databases to a connected and authorized user (see Renner, Abstract). The community server of Renner may provide the access and management facilities to users by sending HTML web pages to the users’ browsers. As is shown in Tables 2 and 4 of Renner, these HTML web pages can be constructed by applying eXtensible Stylesheet Language Translation (XSLT) files to XML components. Thus, the database interface may be specified in XML, or the database may answer queries in XML, and the community server of Renner may apply XSLT files to such XML components to generate HTML web pages to send to user browsers, as is known in the art.

Accordingly, Renner merely discloses the use of XSLT to transform XML files into HTML files, as is well known in the art. Nowhere in Renner does one find anything reading on a data processing cell specification, much less first and second data processing cell specifications, “unnested with respect to each other,” with each data processing cell

specification having a plurality of statements including a formula specifying an action or computation, the first data processing cell having a data dependency on the second data processing cell, and specified in a manner to be analyzed before the second data processing cell.” Further, because of its reliance on the nested, ordered tags of XSLT, Renner does not teach “analyzing in real time, the first and then the second data processing cell specification to determine execution order of the actions/computations specified by the first data processing cell specifications, based at least in part on interaction or computation references between the actions or computations specified.”

The Examiner, on pages 9-10 of the final Office Action, cites 3 pairs of XSLT lines from Table 4 of Renner as reading on the first and second data processing cell specifications: lines 33 and 34, lines 36 and 37, and lines 39 and 40. Not one of these pairs, however, has first and second cell specifications that are unnested with respect to each other, as is required by amended claim 1. The “<xsl: value-of />” tags, such as those shown in lines 34, 37, and 40 of Table 4 (which the Examiner reads on the second data cell processing specification), are always nested within the “<xsl:variable />” tags, such as those shown in lines 33, 36, and 39 of Table 4 (which the Examiner reads on the first data cell processing specification). Accordingly, these pairs of XSLT lines simply cannot read on first and second data processing cell specifications that are unnested with respect to each other.

While the XSLT lines of Table 4 do show data processing cell specifications that are unnested with respect to each other, none of these read on “a first and a second data processing cell specification, unnested with respect to each other, specifying a first and a second data processing cell respectively, with each data processing cell specification having a plurality of statements including a formula specifying an action or computation, the first data processing cell having a data dependency on the second data processing cell, and specified in a manner to be analyzed before the second data processing cell” (emphasis added), as is claimed in claim 1. Renner simply does not teach unnested data processing cell specifications having data dependencies on each other.

Even if one were to assume for the sake of argument that Renner discloses or suggests first and second data processing cell specifications, unnested with respect to each other, having data dependencies on each other (a point with which Applicants strongly disagree), Renner does not teach “analyzing ... the first and then the second data processing cell specification to determine execution order” of the specified actions/computations. Rejections under 35 U.S.C. §102 require that the reference explicitly or inherently disclose each and every limitation of the rejected claims. Nowhere in Renner does one find any analysis of cell specifications for the purpose of determining execution order of those cell specifications and their actions/computations. Further, by utilizing the ordered tree structure of XSLT, including nesting of data processing cells, Renner teaches away from determining execution order based on analysis of the first and second cell specifications. The ordered, tree structure-based method of execution order applies to each and every specification regardless of the interactions/computations of the cell specifications of each specification.

The Examiner argues on page 11 of the final Office Action that determining an execution order is inherent to a parsing engine. Applicants readily concede that general point. However, as to the determining recited in claim 1, Applicants reiterate their strong disagreement. As recited in claim 1, the determination of execution order is based on analysis of the first and second data processing cell specifications. Thus, to anticipate the determining recited by claim 1, Renner would need to disclose that the determination of execution order is dependent on an analysis of cell specifications. Renner discloses no such determination. Rather, the execution order in Renner is strictly dependent on the ordered tree-structure of the specification, and is the same for every specification, regardless of the contents of the cells of the specification and their actions/computations. Such a determined execution order is in no way based on an analysis of cell specifications.

Accordingly, amended claim 1 is patentable over Renner under §102.

Amended claims 11 and 21 recite similar limitations to those recited in amended claim 1. Accordingly, for at least the same reasons, claims 11 and 21 are patentable over Renner under §102.

Claims 2-6, 8-10, 12-16, 18-20 depend on either claim 1 or 11, incorporating their limitations respectively. Accordingly, for at least the same reasons, claims 2-6, 8-10, 12, 16, 18-20 are patentable over the Renner under §102.

CLAIM REJECTIONS UNDER 35 U.S.C. § 103

In “Claim Rejections – 35 USC § 103” item 6 on page 7 of the above-identified final Office Action, claims 7 and 17 have been rejected under 35 U.S.C. § 103(a) as being obvious over Renner as applied to claims 1 and 11 in view of W3C publications “XML Path Language (XPath) Version 1.0” (hereinafter “XPath”) and “XSL Transformations (XSLT) Version 1.0” (hereinafter “XSLT”) that are purportedly stable documents published as W3C recommendations on 16 November 1999. For at least the reasons previously provided, Applicants respectfully disagree.

XPath and XSLT, alone or in combination, do not remedy the above-discussed deficiencies of Renner. Therefore, amended claims 1 and 11 remains patentable over Renner, XPath, and XSLT, alone or in combination, under 35 U.S.C. §103(a).

Claims 7 and 17 depend on claims 1 and 11, incorporating their limitations respectively. Therefore, for at least the same reasons, Claims 7 and 17 are patentable over Renner, XPath, and XSLT, alone or in combination, under 35 U.S.C. §103(a).

ALTERNATE REFERENCES

In “Response to Arguments” item 7 on page 12 of the above-identified final Office Action, the Examiner maintains that a prior ground of rejection, *Bex et al*, “A Formal Model for an Expressive Fragment of XSLT”, First International Conference of Computational

Logic, London, July 2000, Proceedings; Springer-Verlag, pp. 1137-1151. (hereinafter "Bex"), can be used as an alternate prior art reference to the above discussed Renner. Applicants respectfully disagree.

Bex is both unavailable and inapplicable, for the ample reasons provided by the Applicants in their April 19, 2006 Appeal Brief. Accordingly, for at least those reasons, Bex does not provide an alternate prior art reference capable of anticipating the present invention as claimed in claims 1-21, or of rendering the invention of claims 1-21 unpatentable, alone or in combination with XPath and/or XSLT.

CONCLUSION

In view of the foregoing, reconsideration and allowance of claims 1-21 are solicited. If the Examiner has any questions concerning the present paper, the Examiner is kindly requested to contact the undersigned at (206) 407-1513. If any fees are due in connection with filing this paper, the Commissioner is authorized to charge the Deposit Account of Schwabe, Williamson and Wyatt, P.C., No. 50-0393.

Respectfully submitted,
SCHWABE, WILLIAMSON & WYATT, P.C.

Date: February 1, 2007

by: /Robert C. Peck/
Robert C. Peck
Reg. No.: 56,826

SCHWABE, WILLIAMSON & WYATT, P.C.
Pacwest Center, Suites 1600-1900
1211 SW Fifth Avenue
Portland, Oregon 97222
Telephone: 503-222-9981